An Enterprise Social Recommendation System for Connecting Swedish Professionals

Nima Dokoohaki, Mihhail Matskin, Usman Afzal and Md. Mustakimul Islam
Information and Communication Technology School (ICT),
Royal Institute of Technology (KTH)
Forum 120, 16440-Kista, Stockholm, Sweden
Email: {nimad, misha, usmana, islam8}@kth.se

***Abstract*—Most cooperative businesses rely on some form of social networking system to facilitate user profiling and networking of their employees. To facilitate the discovery, matchmaking and networking among the co-workers across the enterprises social recommendation systems are often used. Off-the-shelf nature of these components often makes it hard for individuals to control their exposure as well as their preferences of whom to connect to. To this end, trust based recommenders have been amongst the most popular and demanding solutions due to their advantage of using social trust to generate more accurate suggestions for peers to connect to. They also allow individuals to control their exposure based on explicit trust levels. In this work we have proposed for an enterprise trust-based recommendation system with privacy controls. To generate accurate predictions, a local trust metric is defined between users based on correlations of user's profiled content such as blogging, articles wrote, comments, and likes along with profile information such as organization, region, interests or skills. Privacy metric is defined in such a way that users have full freedom either to hide their data from the recommender or customize their profiles to make them visible only to users with defined level of trustworthy.**

I. Introduction

Social recommendations [1], have become very effective tools for friends and peers to discover each other and maintain relationship on the networks. Recent studies have shown that "People you may know" feature on LinkedIn is used by more than 65 percent of the site users [1]. Humans tend to have different preferences and within social networking context such differences can be captured and interpreted correctly. In order to achieve this recommendation systems must be equipped with personalization heuristics that are used to process, filter, and display available information in such a manner that adheres to each individuals interaction behavior. Often recommendation components used for creating such models of users are off-the-shelf components and do not have enough flexibility for proper adaptation to users behaviors. Such adaptation is especially important when context of recommender system varies [2]. Moreover, additional behavioral features are needed both on functional [3] and on interface levels to allow users properly know the matches being suggested to them both through interactive interfaces [4] and through preference controls. Within this work, we have proposed for a recommendation system that is capable of suggesting similar users based on several behavioral and profile content factors. The engine estimates trust between two people based on implicit and explicit relations between users and recommends a set of users that system believes share similar tastes. In order to alleviate the privacy concern, system uses privacy controls. It provides individual with possibility to decide whether their profiles are going to be used in recommendation generation process or not. Also privacy controls allow the users restrict access to their profiles by different levels of users. In addition the engine also tries to explain the reason for recommendation. Our engine is capable of fine-tuning suggestions based on various constraints (*correlations*). This allows us to infer implicit trust values depending on different criteria such as shared regions, belonging to the same organization, having similar interest, skills, comments on blog posts etc. We have built the recommendation system on the top of a dataset extracted from a Swedish networking platform that is used for matchmaking and introducing professional innovators of the Nordic region. Within the experimental part we present results of our experiments with generating user-specific recommendations considering variations of trust and privacy factors. The rest of this paper is divided into the following sections: First a comprehensive background is presented. This is followed by the framework description. Then experimental part is outlined, followed by conclusion remarks and future works.

II. Related Work

A. Social Recommendation Systems within Enterprise

As employees of organizations join on-line communities and contribute to content, new opportunities for analysis are introduced while the growing volume of data introduces new challenges. In order to stay in touch with colleagues, employees use social networking systems. They are motivated by possibilities of getting connections with more co-workers on a personal level, advancing their career within the company, and recruiting colleagues for their ideas [5]. Social Recommender Systems (SRSs) aim to address this information overload by presenting the most attractive and relevant content to users [6], [7]. SRS is an umbrella term for personalized recommendation techniques which can recommend various types of social data including content (blogs, wikis, etc.), tags, people, and

[1] http://www.forbes.com/sites/cherylsnappconner/2013/10/27/five-linkedin-strategies-you-havent-thought-of-before/

communities. Widely utilized examples of SRS include the recommendation systems implemented at LinkedIn [8], [9].

Guy et al. [6] propose for a social recommendation system that identifies strangers who share similar interests. The aim is introducing new people to the user, in contrast to exploring and searching among the neighbors. Authors focus on connecting strangers (unfamiliar co-workers) within the organization. Our approach also helps end-users to get recommendations from strangers based on similar attributes and tastes. However, people often are not interested to get a friend request from the people whom they do not know at all. They often like to connect with friends of friends to whom they share the same interest. Moreover, our approach also finds out similarity when people did some action on common content such as commenting or liking the same blog post.

In another related work Guy et al. [7] build social recommendations based on people and tags. Relationship information among people, tags, and items, is aggregated across different sources within the enterprise and based on the collected content, system recommends items from people and tags, related to the user. The work highlights the value of tags as accurate content descriptors that take into account human perceptions of the content. However, they did not use any explicit input to the system such as rating, likes etc. Our system highly depends on explicit trust value among the neighbors and this makes it more reliable to the enterprise users. Tagging is normally used as a free text in most of the systems and it does not always reflect what users want. Within our system we also use user-tag relationship, but with lower priority than other constraints. We also deal with cold start problem when having new users.

B. Exploiting Behavior in Enterprise Recommendation

While modeling behavioral traits has been a research agenda for recommendation research [10] business examples of certain traits such as privacy and trust are limited. It has been shown that people primarily connect to individuals they already know, and less likely approach strangers to initiate and maintain a connection [11]. This motivates us to use a notion of social trust which should help users find their trustworthy friends and share their preferences with each other. While trust plays crucial role in many research areas such as psychology, philosophy, sociology and computer science it is still difficult formally define this notion. We use the following simple explanation: humans believe in something or someone based on their knowledge and experience and when their belief reaches a certain level of confidence it becomes a trust [12], [13]. Collaborative filtering (CF) [14], [15] generates the recommendations based on similarity between users. However, similarity measurement may be not sufficient enough when user profiles are sparse. Research has shown that there is a correlation between similarity and trust [14]. The more users are similar the more they can trust each other. Hence, trust can be considered as a measure for establishing a relationship between two users in recommendation systems. In our system, we have defined trust between two users as a measure computed using different constraints (as explained later) based on either user activities (blog writing, liking or commenting) or user profiles (organization, region, interests or skills). The more information about user profiles and activities the system knows the more precise is the trust measure. However due to the huge exposure of personal information, nowadays the challenge is to design effective privacy mechanisms that protect user's information against unwanted usage. The objective of designing privacy scheme is ensuring that a user's on-line information is disclosed only to sufficiently trustworthy parties [15]. Different social networks use different trust models that exploit the underlying inter-entity trust information [16].

In our work, we have defined the privacy controls in such a way that users have full freedom to hide their data from the recommendation process [17]. It enables them to keep privacy during their presence in the network. Along with that, they also have the option to customize their profiles to be visible according to different levels of trust they define in their profile settings.

III. Recommendation Framework: Computational Flow and Component Details

A. Execution Flow

Our system recommends users based on different *correlation* functions used for computing trust as a weighted average of these correlation functions. Figure 1 shows the control and data flow for user log in scenario. Recommendation engine executes this flow when a user visits the site and logs in with its account.

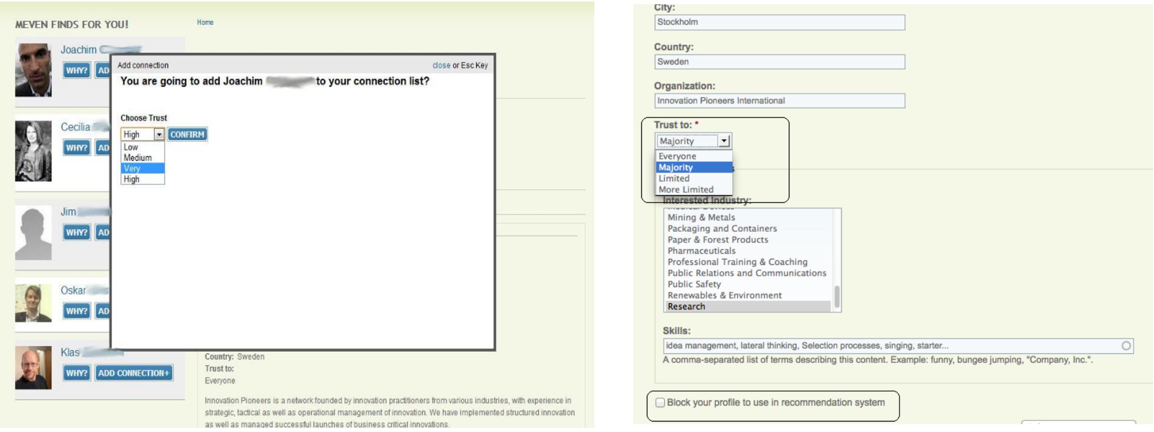
B. Trust: Measuring the Strength of Relationships

Within the framework *Implicit Trust* is a computed measure of trust between users, obtained by measuring the correlation between users. The value of implicit trust is generated by the system. We have also defined *Explicit Trust* as a measure by which a user can explicitly define the acceptable level of trustworthiness towards other users. Figure 2 depicts the user interface for specifying explicit trust levels. The idea is that by choosing a level of trust a user can specify to whom it allowed be recommended. The table below shows the defined explicit trust levels. Explicit Trust is used by the system when recommending a source user (truster) to current user (trustee) if the defined *explicit trust* of truster is equal or greater than the implicit trust value computed by the system.

Trust Level	Explanation
Level 4	Be recommended to everyone
Level 3	Be recommended to majority of users
Level 2	Be recommended to limited users
Level 1	Be recommended to a few users

C. Privacy: Hiding Profiles During Calculations

We have defined a privacy mechanism in such a way that users have full freedom to exclude themselves from the recommendation process. Such privacy protection technique has become popular recently following the revelations of third


MEVEN FINDS FOR YOU!

Home

Joachim [redacted]

WHY? AD

Cecilia [redacted]

WHY? AD

Jim [redacted]

WHY? AD

Oskar [redacted]

WHY? AD

Klas [redacted]

WHY? ADD CONNECTION+

Add connection — close or Esc Key

You are going to add Joachim [redacted] to your connection list?

Choose Trust

High ▾ CONFIRM

Low
Medium
Very
High

Country: Sweden
Trust to:
Everyone

Innovation Pioneers is a network founded by innovation practitioners from various industries, with experience in strategic, tactical as well as operational management of innovation. We have implemented structured innovation as well as managed successful launches of business critical innovations.

City:
Stockholm

Country:
Sweden

Organization:
Innovation Pioneers International

Trust to: *
Majority ▾

Everyone
Majority
Limited
More Limited

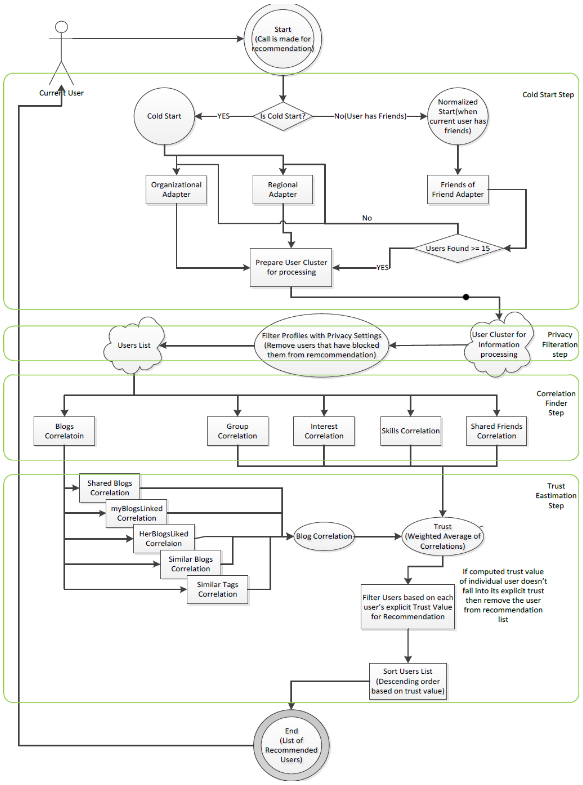
Interested Industry:

Mining & Metals
Packaging and Containers
Paper & Forest Products
Pharmaceuticals
Professional Training & Coaching
Public Relations and Communications
Public Safety
Renewables & Environment
Research

Skills:
idea management, lateral thinking, Selection processes, singing, starter...
A comma-separated list of terms describing this content. Example: funny, bungee jumping, "Company, Inc.".

☐ Block your profile to use in recommendation system

Fig. 2. Behavioral customizations: specifying explicit trust levels while receiving recommendations (left), and customizing privacy controls in profile (right).



Current User
Start (Call is made for recommendation)
Cold Start Step
Cold Start
YES
Is Cold Start?
No(User has Friends)
Normalized Start(when current user has friends)
Organizational Adapter
Regional Adapter
Friends of Friend Adapter
No
Users Found >= 15
YES
Prepare User Cluster for processing
Users List
Filter Profiles with Privacy Settings (Remove users that have blocked them from remcommendation)
User Cluster for Information processing
Privacy Filteration step
Correlation Finder Step
Blogs Correlatoin
Group Correlation
Interest Correlation
Skills Correlation
Shared Friends Correlation
Trust Eastimation Step
Shared Blogs Correlation
myBlogsLinked Correlation
HerBlogsLiked Correlaion
Similar Blogs Correlation
Similar Tags Correlation
Blog Correlation
Trust (Weighted Average of Correlations)
If computed trust value of individual user doesn't fall into its explicit trust then remove the user from recommendation list
Filter Users based on each user's explicit Trust Value for Recommendation
Sort Users List (Descending order based on trust value)
End (List of Recommended Users)

Fig. 1. Data and control flow for user log in scenario.

party tracking of users on the Web [17]. We state that any individual should have freedom to hide or block completely or partially itself from any data mining process. Such feature enables the users to blur their presence in the network. Users should also have the option to customize their profiles to be visible according to the different levels of trust, as seen in Figure 2.

IV. Enterprise Information Sources For Recommendations

As it was mentioned earlier several information sources are mined in application to find similarity and trust. Within this section brief details of each information source are presented.

A. *Organization and Regional Data Sources*

During the *cold start* (when user does not have any friends or content), the system uses *organization* source. It finds out neighborhood cluster (a cluster of users who the system finds closer to the current user) based on workplace (company or employer) of the current user. System also will use *regional* source to find another neighborhood cluster based on region (address, area or country) of the current user.

B. *Affiliation Network Data Sources*

This source is used when current user has friends. System will start from the friends having high trust value for the current user and retrieves a user social graph. The system will continue building a list of users until one of the following conditions are satisfied: 1) Threshold (*k* users in our settings) has reached, 2) System has traversed the social graph of the current user.

V. Correlation functions

We define *correlation functions* as heuristics that compute similarity between two users. All the correlation functions are used when we have a neighborhood cluster either after a cold start, or during the normal execution when the current user already has friends.

A. *Interest Correlation Function (Interest Similarity)*

This correlation function computes correlation between two users based on the interests they have mentioned in their profiles. *Interest* is a pre-defined field in the user profile. We use the similarity matching algorithm below for computing the interest function. In this algorithm $Cor_I(u, v)$ is correlation between users u and v, $Norm(MatchedInterestCount)$ is the function that normalizes the integer into the range$[1, 4]$ and $I(u)$ and $I(v)$ are interests of the users u and v.

Algorithm 1 Interest Correlation

```
$MatchedInterestCount = 0$
**for all** interest in $I(u)$ **do**
    **if** I(v).Contains(interest) **then**
        $MatchedInterestCount++$
    **end if**
**end for**
$Cor_I(u, v) = Norm(MatchedInterestCount)$
```

B. Skills Correlation Function (Tag Distance)

Skills Correlation function computes similarity between two users based on skills they have mentioned in their profiles. In our system the skills are declared tags and we chose *Damerau-Levenshtein distance* [18] for tag matching. Originally the algorithm computes the distance between two tags by adding/replacing/removing characters to match them. However in our settings we have used the algorithm in a slightly different way and compute distance of each tag of the current user to all tags of the user being compared. This allows finding the minimum distance per each tag. In the Algorithm 2 $distance_{DM}$ represents the Damerau-Levenshtein distance function, Ui_{Tag} represents a single tag of user i and Ui_{Tags} refers to all tags of that user.

Algorithm 2 Skills Correlation

```
$tagMatchedCount = 0$
**for** $U1_{Tag}$In $U1_{Tags}$ **do**
    $d = Min(distance_{DM}(U1_{Tag}, U2_{Tags}))$
    $U1_{Tag}.distance = d$
    **if** $(U1_{Tag}.distance) \leq n$ **then**
        $tagMatchedCount++$
    **end if**
**end for**
```

C. Community Memberships Correlation

Community Memberships correlation function computes the similarity between two users based on their memberships in different groups. In our system the groups have parent child relationship. The correlation function does direct matching where it tries to find if two users are members of a similar group. It also does indirect matching where the function tries to find relationship between the groups and then computes similarity between the members. The relationship between the groups is maintained in a separate table that helps in finding the parent-child relationship between the groups.

D. Friends Correlation Function

This is a simple correlation based on shared friends between two users. The function computes the correlation by comparing friends of two users, and returns the normalized value. It is a matching measure of each pair of users in the friend lists.

E. Blog Correlation Functions

This function computes blog correlation between two users by applying different filter functions. The idea is to find taste of two users based on their blog activities (liking, commenting, categorizing and tagging). This shows how likely two users will add each other to their social networks. The following model is used to compute the blog correlation between two users using different filters:

$$\begin{gathered} BlogCor(u, v) = \sum_{i=1}^{n} (Blog(u, v)_{shared} + \\ Blog(u) + Blog(u_{neighbour}) + \\ Sim(u, v)_{blogs} + Sim(u, v)_{tags})/n \end{gathered}$$

where $Blog(u, v)_{shared}$ is the blogs shared between users being compared, $Blog(u)$ is the interested blogs of current user, $Blog(u_{neighbour})$ is the blogs of interest of current users' neighbor, $Sim(u, v)_{blogs}$ is the similar blogs that both users have in common and $Sim(u, v)_{tags}$ is the similar tags that both users have in common. In more details it can be explained as follows:

1) Shared Blogs: This function calculates the number of blogs that are shared between users u and v. The idea is to find the similar taste between the users by looking onto their trends of Liking or Commenting blogs in social networks.

2) Neighbor Interested Blogs: This is a special function from the perspective of recommending someone to current user based on current user's explicit interest towards the recommended user blogs. The overall idea of the recommendation system is to figure out how likely a user will be interested in another user. The result will be highly affected if the user has explicitly shown interest in other users.

3) Similar Tags and Blogs: Tags are extensively used in social systems for tagging object (documents, profiles etc.) and are normally reused by the users. In our system users tag their blogs or other blogs with appropriate keywords. We used the same Damerau-Levenshtein distance for comparing the skills.

Similar blogs function computes the overall similarity between the blogs of two users and it mainly looks for the meta-data of the blog. In our system blogs have such categories as a meta-data field for identifying the field of interest for the blog. The category is a choice field with parent-child relationship to other categories.

4) Overall (Weighted) Blog Correlation: Having each sub-function in Blog correlation we compute the overall blog correlation as a weighted average of individual correlation functions as discussed above. The following formula is used for finding weighted Blog correlation between each pair of users:

$$BlogCor(u,v)_{Weighted} = \sum_{i=1}^{n}(\alpha Blog(u,v)_{shared} + \beta Blog(u) + \mu Blog(u_{neighbor}) + Sim(u,v)_{Blogs} + Sim(u,v)_{Tags})/n$$

where μ, β, and α are weights for the associated correlations. Rests of the parameters are the same as for the blog correlation formula. Because of the interested blogs by neighbors indicate the explicit interest of current user in the recommended user we keep the weights of this blogs higher than weights of other blogs in the system.

VI. Experiments

A. Dataset Description and Evaluation Methodology

Evaluation of recommendation systems has been subject to extensive research. A large number of real-world recommendation systems are often developed by experimenting with offline data. If necessary, *randomly sampling users and items* are considered as a preferable method for reducing data, although this can introduce other biases into the experiment (e.g. this might favor approaches that work better with more sparse data) [19]. To create a user-centric evaluation scheme, we have created a set of test user profiles for our experiments. These profiles are represented with the following identifiers: $UID = \{1, 506, 508, 509, 510\}$. The user selection has been completely randomly sampled. The data we have used for our system represents a social network of highly skilled business related professionals [2], who work together to bring innovative ideas. The system contains profiles of professionals from different departments and companies and they are very much interested in knowing about other professionals across the enterprise landscape. The users mainly have sales background, they belong to management departments or they even are the executives of the companies. The dataset contains about 550 user profiles along with more than 3000 blog posts, articles, tags, categories, groups, etc. The profiles in the system contain information about region, organization, education, interests, skills, social activity feeds (blogs, comments, linking, and articles) and friends of a user.

B. Influence of Privacy Metric on Recommendations

In this part of experiment we aimed to observe how privacy controls affects recommendation quantity, as well as the average trust value. We have used the same data as above and perform tests both during cold-start and normal flow of execution. As we already mentioned, we have two kinds of privacy controls. We have performed experiments to evaluate the case where users may hide themselves in order not being considered during the recommendation process.

The results in Figure 3 visualize the trend while varying the number of hidden users. The results show the number of recommendations in both cold-start and normal flow for a particular setting of hidden users. These results show recommendations size for different users while increasing the number of users who hide themselves from recommendation process. It is observed that as we keep increasing the number of hidden users our recommendations size become affected. For instance, the number of recommendations generated for user (*UID-509*) is 55 when 50 users have hidden themselves from the recommendations process. This number goes down to 25 when 250 users were hidden. The results indicate that the reliability of our recommendation system is immensely dependent on the involvement of the users in the system.

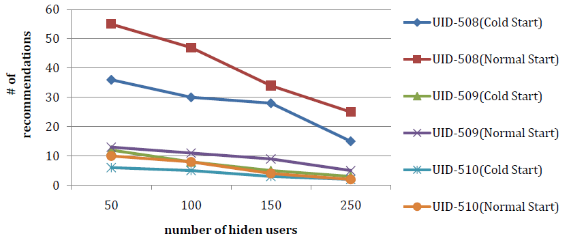


Fig. 3. Influence of privacy on the recommendation quantity.

C. Influence of Implicit Trust on Recommendations Quality

In this experiment we evaluated the dependencies and influence of different correlation functions to the generated recommendations. As shown previously, we have used different correlation functions to compute trust between two users. In the results visualized in Figure 5, average trust values of recommendations per a single user with individual correlation functions is shown at the top. It is noted that individual correlations functions produce high trust values when used alone. While observed values are derived for each activity, we believe the accuracy of combined values gives a better understanding of implicit relations among users. We would also like to notice that the recommendations are more accurate when different correlations are combined together. In the second plot we have shown the results of average trust values produced when different correlation functions were used in a combined manner. We can see that average trust values have decreased compared to the top figure but these recommendations are most likely be interesting for the current user, as they reflect combination of activities. The green line (where all correlation functions are used) shows that the average trust value remains almost the same independently on increase the number of users while the others lines tend to decrease when increasing number of recommended users. This also indicates that a mix of several factors as an implicit trust enabler is a more stable measure, as of compared to isolated factors. Figure 4 depicts the variations of implicit trust in different flows of recommendation process.

Conclusions and Future Directions

In this work, we have implemented an enterprise social recommendation system that takes into account behavioral

[2] Innovation Pioneers. http://www.innovationpioneers.se

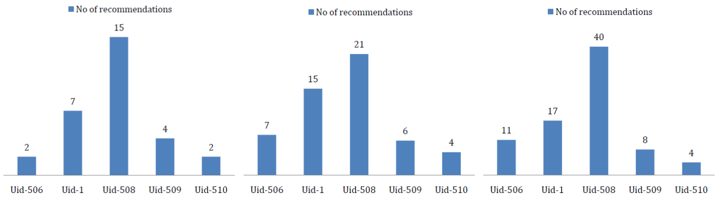


Fig. 4. Influence of explicit trust on recommendations size. left to right: variations of recommendations ratio with low, mid and high trust levels.

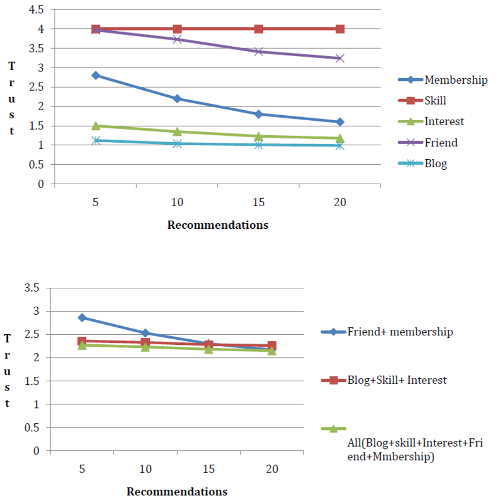


Fig. 5. Impact of Trust on Correlation functions: recommendations received with respect each correlation model (top) and correlation models combined (bottom).

traits such as trust based on profiles, network and various contents shared. We also include privacy metrics to control the level of exposure of user information towards other users depending on their profiles settings. We examined our system based on a snapshot of the system at the time of study. In our evaluation part, using a user-specific evaluation technique, we presented results of recommendation with respect to various trust and privacy settings using different measures. We plan to further analyze the recommendation system with new sets of users as well as study the performance from system-centric point of views.

References

[1] I. Guy and D. Carmel, "Social recommender systems," in *Proceedings of the 20th International Conference Companion on World Wide Web*, ser. WWW '11. New York, NY, USA: ACM, 2011, pp. 283–284.

[2] G. Adomavicius and A. Tuzhilin, "Context-aware recommender systems," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Springer US, 2011, pp. 217–253.

[3] N. Dokoohaki, "Trust-based user profiling," Ph.D. dissertation, KTH, Software and Computer systems, SCS, 2013, qC 20130219.

[4] B. P. Knijnenburg, M. C. Willemsen, Z. Gantner, H. Soncu, and C. Newell, "Explaining the user experience of recommender systems," *User Modeling and User Adapted Interaction*, pp. 1–64, 2012.

[5] J. DiMicco, D. R. Millen, W. Geyer, C. Dugan, B. Brownholtz, and M. Muller, "Motivations for social networking at work," in *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work*, ser. CSCW '08. New York, NY, USA: ACM, 2008, pp. 711–720.

[6] I. Guy, S. Ur, I. Ronen, A. Perer, and M. Jacovi, "Do you want to know?: Recommending strangers in the enterprise," in *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work*, ser. CSCW '11. New York, NY, USA: ACM, 2011, pp. 285–294.

[7] I. Guy, N. Zwerdling, I. Ronen, D. Carmel, and E. Uziel, "Social media recommendation based on people and tags," in *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval - SIGIR '10*. New York, New York, USA: ACM Press, Jul. 2010, p. 194.

[8] M. Amin, B. Yan, S. Sriram, A. Bhasin, and C. Posse, "Social referral: leveraging network connections to deliver recommendations," in *Proceedings of the sixth ACM conference on Recommender systems*. ACM, 2012, pp. 273–276.

[9] C. Posse, "Key lessons learned building recommender systems for large-scale social networks," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '12. New York, NY, USA: ACM, 2012, pp. 587–587.

[10] T. Iwata, K. Saito, and T. Yamada, "Modeling user behavior in recommender systems based on maximum entropy," in *Proceedings of the 16th International Conference on World Wide Web*, ser. WWW '07. New York, NY, USA: ACM, 2007, pp. 1281–1282.

[11] R. K. Chellappa and R. G. Sin, "Personalization versus Privacy: An Empirical Examination of the Online Consumers Dilemma," *Information Technology and Management*, vol. 6, no. 2, pp. 181–202, Apr. 2005.

[12] R. Falcone and C. Castelfranchi, "Social trust: A cognitive approach," in *Trust and Deception in Virtual Societies*, C. Castelfranchi and Y.-H. Tan, Eds. Springer Netherlands, 2001, pp. 55–90.

[13] S. Marsh and P. Briggs, "Examining trust, forgiveness and regret as computational concepts," in *Computing with Social Trust*, ser. HumanComputer Interaction Series, J. Golbeck, Ed. Springer London, 2009, pp. 9–43.

[14] S. Magureanu, N. Dokoohaki, S. Mokarizadeh, and M. Matskin, "Epidemic Trust-based Recommender Systems," in *IEEE international conference on Social Computing 2012 (SocialCom12)*. Amsterdam, Netherlands: IEEE Computer Society, 2012, pp. 461–470.

[15] R. Bunea, S. Mokarizadeh, N. Dokoohaki, and M. Matskin, "Exploiting Trust for Privacy Inference in a Collaborative Filtering Recommender Framework," in *PinSoDa: Privacy in Social Data, in conjunction with the 11th IEEE International Conference on Data Mining (ICDM 2012)*, A. Monreale and D. Pedreshi, Eds. Brussels, Belgium: IEEE Computer Society, 2012.

[16] N. Li, M. Najafian Razavi, and D. Gillet, "Trust-aware privacy control for social media," in *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems - CHI EA '11*, 2011, p. 1597.

[17] J. Mayer and A. Narayanan, "Do not track: Universal web tracking opt-out," *Center for Internet and Society, Stanford Law School, Stanford, California (donnottrack.us)*, 2011.

[18] L. Yujian and L. Bo, "A normalized levenshtein distance metric," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 6, pp. 1091–1095, June 2007.

[19] G. Shani and A. Gunawardana, "Evaluating recommendation systems," *Recommender Systems Handbook*, pp. 257–297, 2011.